International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 4 | ISSN : 2456-3307

A Novel Approach for Data Sharing Securely in Cloud Computing

Ch. Radhika Devi[1], K. Jaya Krishna[2]

[1]Department of MCA, QIS college of Engineering and technology, Ongole, Andhra Pradesh, India

[2]Associate professor, Department of MCA, QIS college of Engineering and technology, Ongole, Andhra Pradesh, India

ABSTRACT

Cloud storage is aservice of clouds that causes, associations to change from building up in-house information stockpiling frameworks to the cloud. The advances that proposed as of late have offered ascend to the ubiquity and accomplishment of cloud computing. Be that as it may, while outsourcing the information and business application to athird party causes the security and protection issues to wind up noticeably a basic concern. Inside an association, information should be shared among various clients with various certifications. Secure sharing of information among a gathering which causes an insider danger from substantial or noxious client is an essential research issue. Current approach gives an answer on previously mentioned issue. Under this approach, the third party is a responsible for security related tasks like encryption, decryption, and key age, get to control, and so on. In any case, there might be a probability that this third party may indicate noxious conduct and causes insider risk. A superior approach ought to give an answer which limits confide in third party while guaranteeing information classification. We propose an approach, in light of two layers of encryption that tends to such necessity. Under our approach, the information owner plays out a lower layer encryption, though the third party plays out an upper layer encryption over the owner scrambled information. A testing issue is the means by which to keep up secrecy of information. To actualize it we moved access control right dissemination task to the owner. The owner sends a key to legitimate clients for encryption and decryption. This guarantees just substantial client will get an entrance to the information.

Keywords: Cloud Computing, Confidentiality, Access control, Secure Sharing, Privacy, Security.

I. INTRODUCTION

Cloud computing is quickly developing because of the provisioning of versatile, adaptable, and on-demand storage and registering services for clients. Cloud computing offers a powerful method to decrease capital use and operational consumption. This financial advantage is a primary driver of the cloud ubiquity. Be that as it may, SECURITY and protection speak to significant worries in the appropriation of cloud innovations for information stockpiling. A way to deal with alleviate these worries is the utilization of cryptography where information are normally encoded before putting away to the cloud. Though cryptography guarantees the privacy of the information against the cloud, when the information are to be shared among a gathering, the cryptographic services should be sufficiently adaptable to deal with various clients, practice the entrance control, and deal with the keys in a compelling way to defend information secrecy. The information taking care of among a gathering has certain extra attributes rather than two-party correspondence or the information taking care of

having a place with a solitary client. The current, withdrawing, and recently joining bunch individuals can turn out to be an insider danger disregarding information classification and protection.

While receiving a cloud for capacity, the loss of control over information and algorithm raises numerous security worries for associations. The loss of control over information and the capacity stage additionally spurs cloud clients to keep up the entrance control over information (singular information and the information shared among a gathering of clients through the general population cloud).The cloud client scrambles the information before putting away to the cloud, this guarantees cloud doesn't take in any data about client's information. The entrance rights are given to various clients by disseminating key utilized for encryption. In any case, this will bring about exorbitant load over clients. By putting athird party in the middle of client and cloud and designating every single operational load to an third party will bring down load from the client. Be that as it may, at the same time there is a probability that third party may indicate malignant conduct. Henceforth there ought to be a way to deal with beat this.

In this paper, we propose an approach named Secure Data Sharing in Clouds through constraining trust in Third gathering/Server those arrangements with the previously mentioned security. It limits confide in third party/server. While appointing some operational load to athird party this approach guarantees information privacy. For this the idea of two layer encryption is utilized where bring down layer encryption is performed by the information owner and upper layer encryption is performed by third party. The owner gives the expert of document access to client by demonstrating key utilized for bring down layer encryption, while encryption or decoding of the record. Henceforth, by holding control over activities back to information owner this approach jelly privacy.

II. SIGNIFICANCE

Approach tended to in depends completely on third party (CS) for security related tasks like encryption, key age, get to control, decoding, and so forth., however there is a plausibility that this third party demonstrates the noxious conduct and cause an insider risk. This approach has displayed an approach to restrain trust level in completely on third party(CS).The bring down layer encryption at owner guarantees that third party won't get immediate access to information. Document get to specialist is given by the owner by conveying key utilized for bring down layer encryption. Regardless of whether athird party gives document to any unapproved clients, they unfit to get to it as they won't ready to get a key for decoding from record owner.

Also, this approach causes less time utilization. In for every client isolate key offers are should have been figured amid encryption and amid decoding unique key should be registered from shares. Additionally for new client incorporation isolate key offers should be ascertained. In our approach just two keys require one for bring down layer encryption and another for upper layer encryption. Henceforth time for key offer age amid encryption, unique key algorithm amid decryption and age of key offers for recently joining individuals get killed.

III. RELATED WORK

Approach tended to in gave an approach to secure sharing of information among clients with various level of benefit. There is an third party in the middle of client and cloud who is in charge of performing security related tasks like key service, encryption, decoding, and access control. Information is encoded utilizing a solitary symmetric key. Two diverse key offers are produced for every client. One offer is given to the client and the other is kept by athird party named Cryptographic server. Client with one offer guarantees security from the insider risk.

Notwithstanding, this approach depends totally on third party, there is plausibility that third party shows pernicious conduct and cause an insider risk. As proposed in, to defeat the insider danger to cloud classification one guard procedure is to hold control back to the owner. Be that as it may, it will cause inordinate load over owner. As Suggested in, the cloud produces the public– private key sets for the greater part of the clients and transmits people in general keys to the majority of the taking an interest clients. Fractional decoding is performed at the cloud. Because of the way that key service and fractional decryption are dealt with by the cloud, client renouncement is less demanding to deal with. In any case, the proposed conspire regards the general population cloud both as a trusted and untrusted substance in the meantime. From a security point of view, it isn't prescribed to move the key age procedure to the common multitenant open cloud condition.

This paper displayed an approach which counters the previously mentioned issue. We have utilized two layer encryption plots as recommended in. The lower layer encryption by information owner guarantees that third party won't ready to get to owner's information. Expert to get to information is given by information owner by exchanging key utilized for bring down layer encryption to the client. This guarantees third party won't ready to give an entrance to any unapproved client. Regardless of whether he does as such that unapproved client won't ready to peruse it without key utilized for bring down layer encryption.

IV. PROPOSED WORK

(1) User registration; (2) File Upload; (3) Download Request; (4) File Download

Figure 1. System architecture of Secure Data Sharing in the Cloud through limiting trust in Third party/server Figure1 shows the basic architecture of "secure data sharing in the cloud through limiting trust in third party/ Server".

This proposed framework will work with four elements as takes after: 1) owner; 2) client; 3) server and 4) cloud. The information owner initially allots a remarkable ID to every client of his documents. The clients at that point enlist with owner by giving his own password. The owner keeps up data about every client in list-User List containing special id and password. The data about clients additionally sends to athird party for capacity. While transferring document to cloud the record owner will perform bring down layer encryption and present the scrambled document. Server in the wake of accepting a record performs upper layer encryption on it. The encoded information at that point in this way gets transferred to the cloud for capacity. The document owner appoints get to right (read/write) on record to clients. The rundown of clients, their entrance right and other data like date from which get to right is legitimate will get sent to the third party server .This data gets keep up there as ACL. ACL is kept up for each document containing record id, client id, date, and get to right. The client who wishes to get to the record sends a download demand to the third party server. The third party server gets the remarkable ID and keyword from the client, subsequent to validating the asking for client it downloads the information document from the cloud. The

information document gets decoded and sent back to the client. Client in the wake of accepting document asks for a key to the owner. The owner gives the key he has used to perform bring down layer encryption. Client unscrambles a record with key he has gotten from the owner. For a recently joining part, the owner doles out another one of a kind id and client at that point enlist with owner. The owner will send the data with respect to this client to the server to keep up it in an ACL. For a leaving part, the record will erased from every single individual table.

V. PERFORMANCE STRATEGIES

5.1 User Registration: This activity is performed by the record owner. The owner of a document will initially relegate a unique ID (ID) to every client of his records. At that point client registers with this extraordinary id with owner by giving his password (PassW). Figure 2 indicates client enlistment activity.

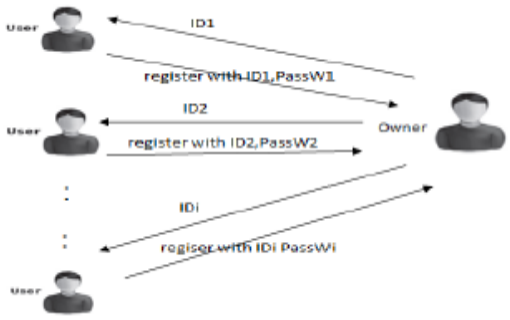

Figure 2. User Registration

5. 2 File Upload

5.2.1 Lower Layer Encryption

Algorithm:

R = {0, 1}^8 // R is any random number

Kf= Hf(R) // after applying hash function the R will completely randomized

F' = SKA(F, Kf)

or each document F separate key Kf is produced. Kf is produced in two stages. In the initial step, the irregular number R of length 8 bits is determined by the record owner. In the subsequent stage, R is gone through a hash work that could be any hash work with a 128-piece yield. The length of Kf is 128 bits. However the length of the key can be changed by prerequisite. The document at that point encoded with a symmetric encryption algorithm utilizing produced key Kf. After fruitful encryption, scrambled document F' gets sent to the server.

5.2.2 Upper Layer Encryption:

Algorithm:

R = {0, 1} ^8/R is any arbitrary number

K'f= Hf (R)/in the wake of applying hash work the R will totally randomize

C = SKA(F', K'f)

The third party server is in charge of upper layer encryption. For this it creates a key(K'f) of length 256 bits. The key K'f is created in two stages. In the initial step, the irregular number R of length 8 bits is determined by the third party server. In the subsequent stage, R is gone through a hash work that could be any hash work with a 256-piece yield. The length of K'f is 256 bits. However the length of the key can be changed by prerequisite. The document at that point scrambled with a symmetric encryption algorithm utilizing produced key K'f. After effective encryption , scrambled record C motivates sent to cloud for capacity. Figure 3 demonstrates a record transfer task which includes both lower layer encryption at owner and upper layer encryption on the server.

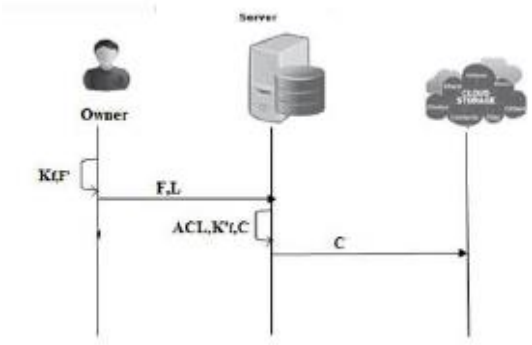

Figure 3. File Upload

5.3 File Download This activity requires adecryption to be performed twice. To begin with at the server and other by asking for client. Figure 4 demonstrates download task.

5.3.1 Decryption at the Third Party Server

Algorithm:

Get IDi and PassWi from the asking for client I. Perform verification and confirm get to right. In the event that confirmation fizzled or get to isn't substantial, at that point Return the entrance denied message to the client.

else

Download C from the cloud.

F' = SKA(C, K'f)

send F' to the client.

end if

At whatever point any client needs to download any record, he/she sends a demand to the third party server. The third party server in the wake of getting a demand confirm client and after effective confirmation download asked for record from the cloud, unscrambles it and sends it to the client.

5.3.2 Decryption at the Requesting User

Client in the wake of getting document from athird party server unscrambles it. For this, the client asks for a key to the document owner. Record owner in the wake of performing effective validation sends a key. The client at that point unscrambles a document with a key he/she got from the owner.

Algorithm:

F=SKA(F',Kf)

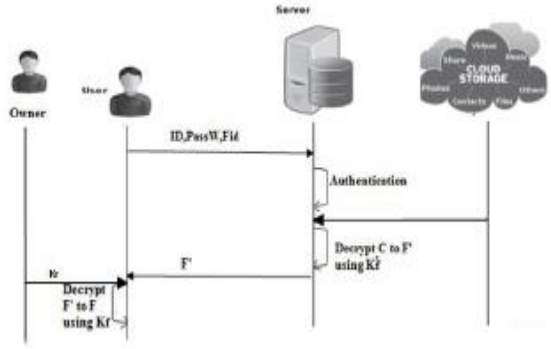

Figure 4: File Download

5.4 File Update This task requires bring down layer encryption to be performed by the client who refreshes the document. For this, client demands for a key to record owner. Record owner subsequent to performing verification sends a key. The client at that point performs encryption on information and sends refresh demand to the server. For encryption, client utilizes symmetric encryption algorithm utilized as a part of algorithm 1. The server performs confirmation and check whether the asking for client has refresh authorization. After a fruitful validation server performs upper layer encryption and transfer record to the cloud. For encryption, server utilizes symmetric key encryption algorithm utilized as a part of algorithm 2. Figure 5 indicates refresh activity.

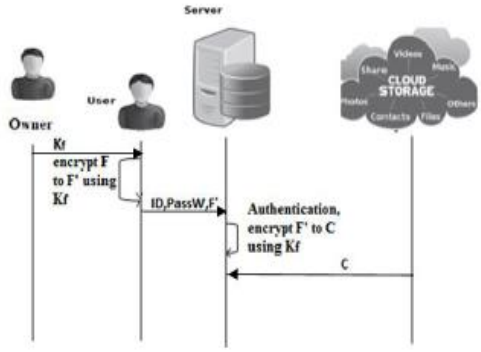

Figure 5. File Update

5.5 New User Inclusion and Departure The incorporation of new client is made by enlisting client with owner by giving new id. This data put away by the owner in the client list. This data likewise gets sent to the third party server for capacity. For recently joined part owner doles out access rights on records. This data additionally gets sent to the third party server to incorporate it into ACL. The data contains document id, client id, date from which get to right is legitimate, get to right. The arrangement of date guarantees in reverse access control. For withdrawing part, the server is advised by the owner. The third party separate at that point erases all records identified with the client from ACL. The withdrawing part won't ready to unscramble information by its own. Henceforth this guarantees forward access control.

VI. PERFORMANCE EVALUATION

6.1 Experimental Setup To assess the performance of the proposed philosophy, we actualized the approach in Eclipse utilizing the JAVA structure. As examined

before, the proposed approach comprises of four substances, the cloud, the third party server, owner and the clients. The cloud benefit which fills in as the cloud server in our usage. The third party and record owner are executed as a server utilizing apache tomcat. This performance takes after the MVC show. Where the view is executed utilizing jsp/html, control is only servlets. The functionalities required by the client that is client level encryption and decoding are executed as a TCPclient application that interfaces with the ownerto get the key. This correspondence is secured by SSL convention. The usefulness that required by the owner to give key is executed as a TCPserver application which constantly kept in running mode. The correspondence between the elements Owner and the third party is refined utilizing URL class (java.net.URL). The class HTTPs URL Connection is utilized to open a safe association with another substance. The plan utilizes the SHA-1 hash work for producing keys at the owner and SHA-256 hash work for creating keys at third party server. The AES for encryption and decoding is utilized. The greater part of the cryptographic tasks like encryption and decoding are executed utilizing a javax crypto Cipher. The class java security. Message Digest is utilized to get to the greater part of the techniques identified with SHA.

6.2 Results We assessed the procedure based on the aggregate time devoured to transfer/download a record to/from the cloud. The aggregate time is made out of the time from the season of accommodation of demand to the CS to the point of time at which the record is transferred/downloaded to/from the cloud. The accompanying circumstances are incorporated into the aggregate time:

1) The key algorithm time at owner and third party server;

2) The encryption/decoding time at owner/client and third party server;

3) The transfer/download time;

4) The season of demand and other related information accommodation to the CS and the cloud.

Fig. 6 demonstrates the outcomes for the transfer time. The greater part of the constituent circumstances is spoken to by discrete line diagrams. The expression "others" alludes to the fourth constituent time examined already. When all is said in done, an opportunity to transfer the information expanded with the expansion in the document estimate. In any case, some negligible changes in time are because of system condition around then. Consequently, the record transfer time was reliant on the system conditions. The key algorithm time is autonomous of record estimate and nearly stayed consistent. The encryption time expanded with the expansion in the record measure. Fig. 7 demonstrates the outcomes for the download activity associated with downloading the record from the cloud and the resulting decoding forms. The pattern of results is comparable as on account of a record transfer. Be that as it may, the circumstances in decoding and the download are changed. There is no compelling reason to process key amid download strategy. Consequently Key algorithm time is wiped out. We have contrasted our technique and the plan introduced. The examination is on the turnaround time for encryption and decoding.

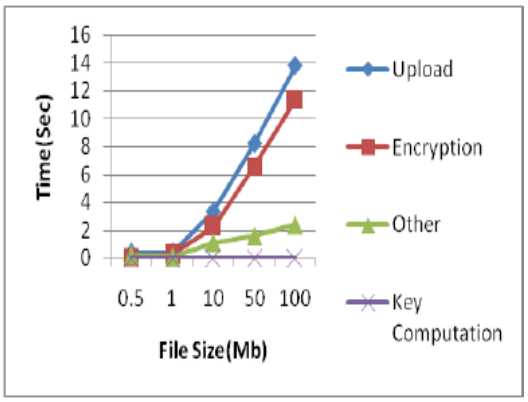

Figure 6(i). Performance of File Uploads

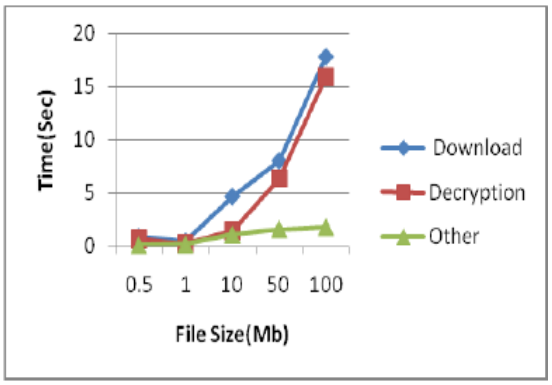

Figure 6(ii). Performance of File Downloads

VII. CONCLUSIONS

We proposed a methodology for secure sharing of information among numerous clients with various certifications. The proposed technique gives information secrecy, secure information sharing without re encryption, and access control for vindictive insiders, and forward and in reverse access control. In addition, proposed procedure tended to issues in past approach and gave its powerful arrangement. The approach gave here can be reached out by fortifying responsibility. Here clients are separated by client id's and keyword. One can give an approach which utilizes distinctive method for guaranteeing responsibility of clients. One can give an elective way to deal with restricting trust in third party/server.

VIII. REFERENCES

[1]. A. N. Khan, M. L. M. Kiah, S. U. Khan, and S. A. Madani, "Towards secure mobile cloud computing: A survey," Future Gen. Comput. Syst., vol. 29, no. 5, pp. 1278–1299, Jul. 2013.

[2]. L. Wei, H. Zhu, Z. Cao, Y. Chen, and A. V. Vasilakos, "Security and pri- vacy for storage and computation in cloud computing," Inf. Sci., vol. 258, pp. 371–386, Feb. 2014.

[3]. Cloud security Alliance, "Security guidelines for critical areas of focus in cloud computing v3.0," 2011.

[4]. D. Chen et al., "Fast and scalable multi-way analysis of massive neural data," IEEE Trans. Comput., DOI: 10.1109/TC.2013.2295806, 2014, to be published.

[5]. A. N. Khan, M. M. Kiah, S. A. Madani, M. Ali, and S. Shamshir-band, "Incremental proxy re-encryption scheme for mobile cloud computing environment," J. Supercomput., vol. 68, no. 2, pp. 624–651, May 2014.

[6]. Y. Chen and W. Tzeng, "Efficient and provably-secure group key man- agement scheme using key derivation," in Proc. IEEE 11th Int. Conf. TrustCom, 2012, pp. 295–302.

[7]. Cloud Security Alliance, "Security guidelines for critical areas of focus in cloud computing v3. 0," 2011.

[8]. Zhifeng Xiao and Yang Xiao, Senior Member,IEEE, "Security and Privacy in Cloud Computing", IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 15, NO. 2, SECOND QUARTER 2013.

[9]. Mazhar Ali, Student Member, IEEE, Revathi Dhamotharan Eraj Khan, Samee U. Khan, Senior Member, IEEE, Athanasios V. Vasilakos, Senior Member, IEEE, Keqin Li, Fellow, IEEE, and Albert Y. Zomaya, Fellow, IEEE,"SeDaSC: Secure Data Sharing in Clouds"IEEE SYSTEMS JOURNAL 2015.

[10]. S. Seo, M. Nabeel, X. Ding, and E. Bertino, "An Efficient Certificateless Encryption for Secure Data Sharing in Public Clouds," IEEE Trans. Knowl. Data Eng., Vol. 26, no. 9, pp. 2107–2119, Sep. 2013.

[11]. Mohamed Nabeel and Elisa Bertino, Fellow, IEEE,Privacy Preserving Delegated Access Control in Public Clouds", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 9, SEPTEMBER 2014.

[12]. L. Xu, X. Wu, and X. Zhang, "CL-PRE: A certificateless proxy re- encryption scheme for secure data sharing with public cloud," in Proc. 7th ACM Symp. Inf. , Comput. Commun. Security, 2012, pp. 87–88.

[13]. P. Gutmann, "Secure deletion of data from magnetic and solid-state memory," in Proc. 6th USENIX Security Symp. Focusing Appl. Cryptog- raphy, 1996, p. 8.
[14]. S. Seo, M. Nabeel, X. Ding, and E. Bertino, "An Efficient Certificate- less Encryption for Secure Data Sharing in Public Clouds," IEEE Trans. Knowl. Data Eng., vol. 26, no. 9, pp. 2107–2119, Sep. 2013.
[15]. Y. Chen, J. D. Tygar, and W. Tzeng, "Secure group key management using uni-directional proxy re-encryption schemes," in Proc. IEEE INFOCOM, pp. 1952–1960.

ABOUT AUTHORS:

CH.RADHIKA DEVI is currently pursuing her MCA in QIS College of Engineering and Technology, Ongole, A.P. Her aria of interest cloud computing .

Mr. K . JAYA KRISHNA he is currently working as an Associate Professor in Master of Computer Applications Department , in QIS College of Engineering and Technology, Ongole , AP. His aria of interest Big data and Data mining. research includes networking and data mining.